**ASSIGNMENT 3**

**DATA STRUCTURE AND ALGORITHM LAB**

**Program 1:**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void display\_from\_end();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Display from end\n4.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

display\_from\_end();

break;

case 4:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void display\_from\_end()

{

node \*ptr;

if(last==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=last;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

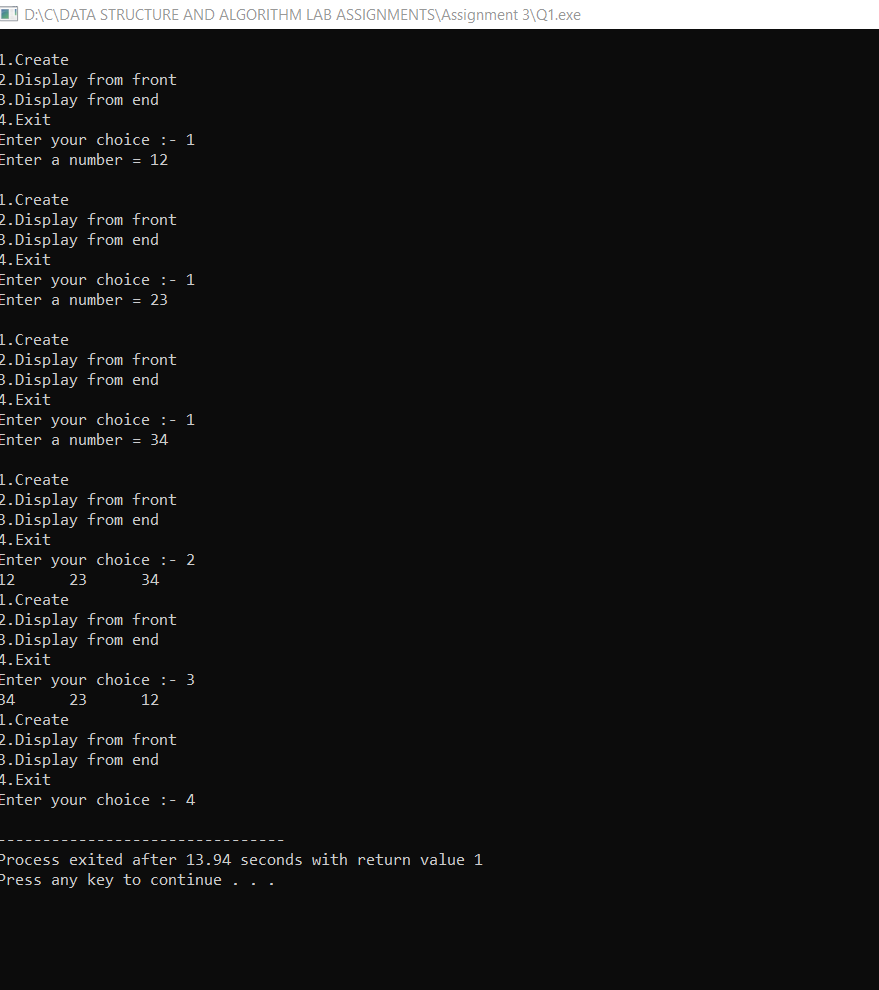
ptr = ptr->prev;

}

}

}

**Output:-**

****

**Program 2:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void display\_from\_end();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Display from end\n4.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

display\_from\_end();

break;

case 4:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void display\_from\_end()

{

node \*ptr;

if(last==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=last;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

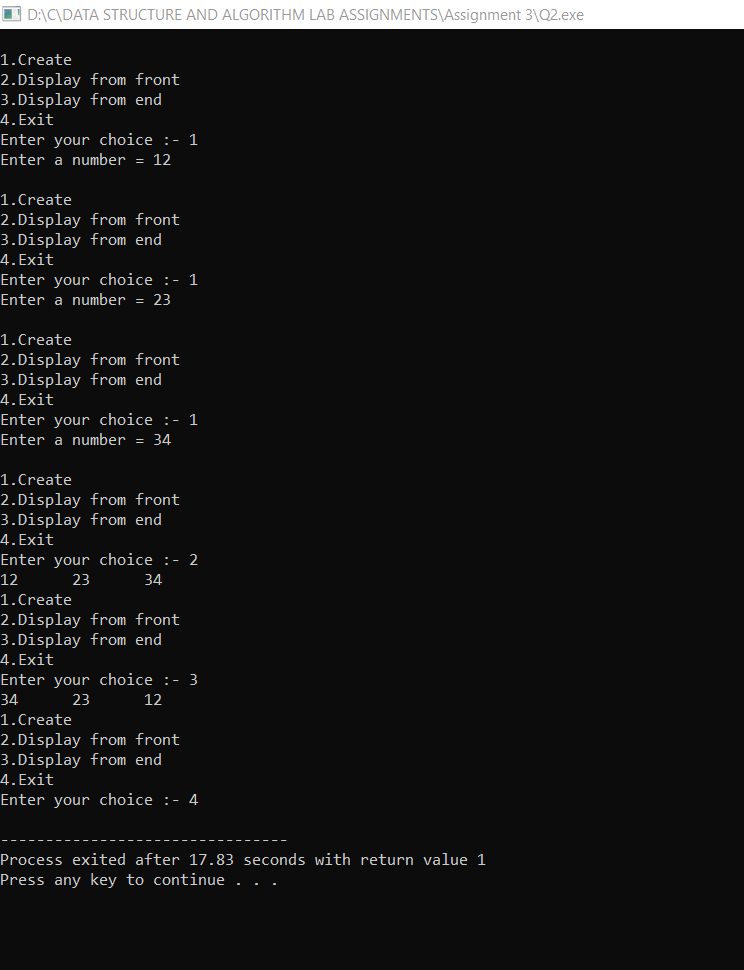
ptr = ptr->prev;

}

}

}

**Output:-**

****

**Program 3:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void insert\_beg();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Insert at beginning\n4.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

insert\_beg();

break;

case 4:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void insert\_beg()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

if(head==NULL)

printf("Linked List does not exists!!!!");

else

{

ptr = head;

nw->next = head;

ptr->prev = nw;

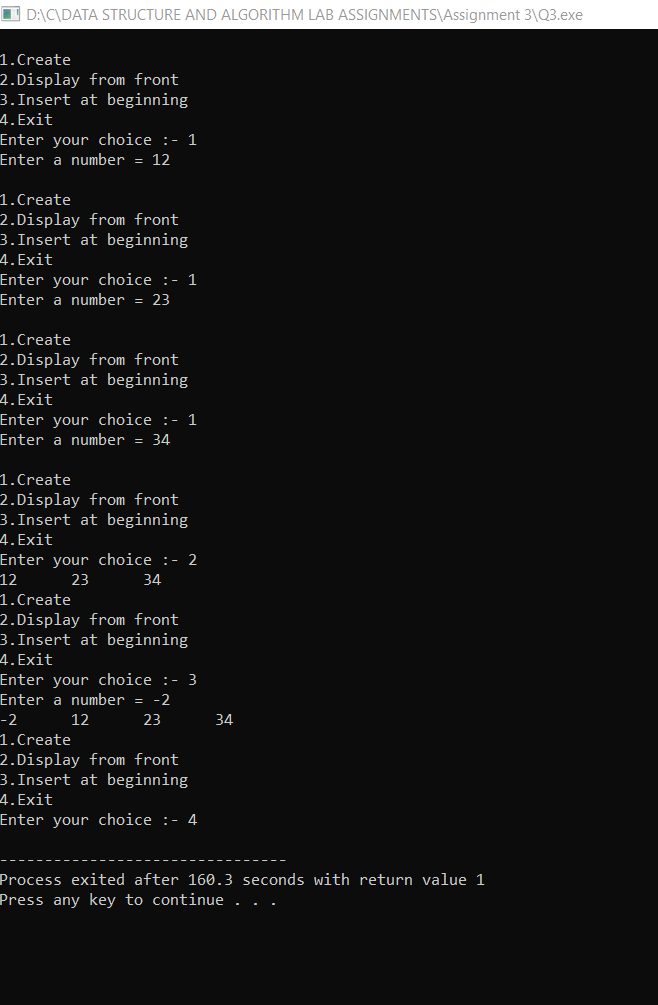
head = nw;

}

display\_from\_front();

}

**Output:-**

****

**Program 4:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void insert\_end();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Insert at end\n4.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

insert\_end();

break;

case 4:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void insert\_end()

{

node \*nw,\*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

if(last==NULL)

printf("Linked list does not exists!! ");

else

{

ptr = last;

ptr->next=nw;

nw->prev = ptr;

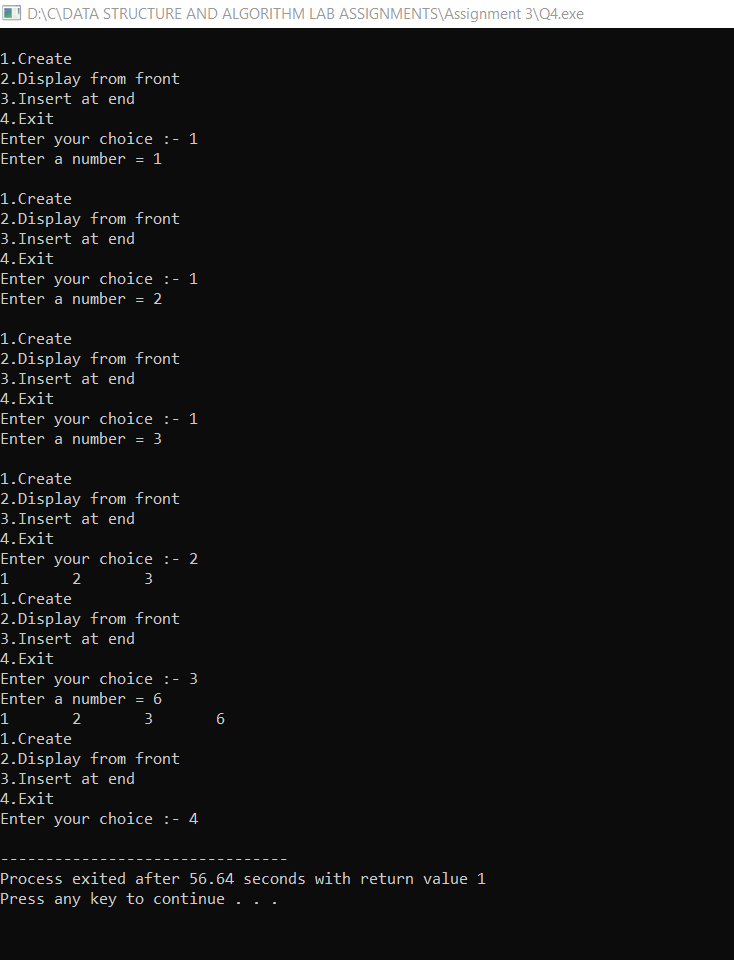
nw->next=NULL;

}

display\_from\_front();

}

**Output:-**

****

**Program 5:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void insert\_at\_any();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Insert at any position\n4.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

insert\_at\_any();

break;

case 4:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void insert\_at\_any()

{

node \*nw,\*ptr,\*ptr1;

int pos,i=1;

fflush(stdin);

nw = (node \*)malloc(sizeof(node \*));

printf("Enter the position where you want to enter = ");

scanf("%d",&pos);

printf("Enter a number = ");

scanf("%d",&nw->data);

if(head==NULL)

printf("Linked list does not exists!!!");

else if(pos==1)

{

ptr = head;

nw->next = head;

ptr->prev = nw;

head = nw;

}

else

{

ptr=head;

ptr1=head;

while(ptr!=NULL&&i!=pos)

{

ptr1=ptr;

ptr = ptr->next;

i++;

}

ptr1->next=nw;

nw->prev=ptr1;

nw->next=ptr;

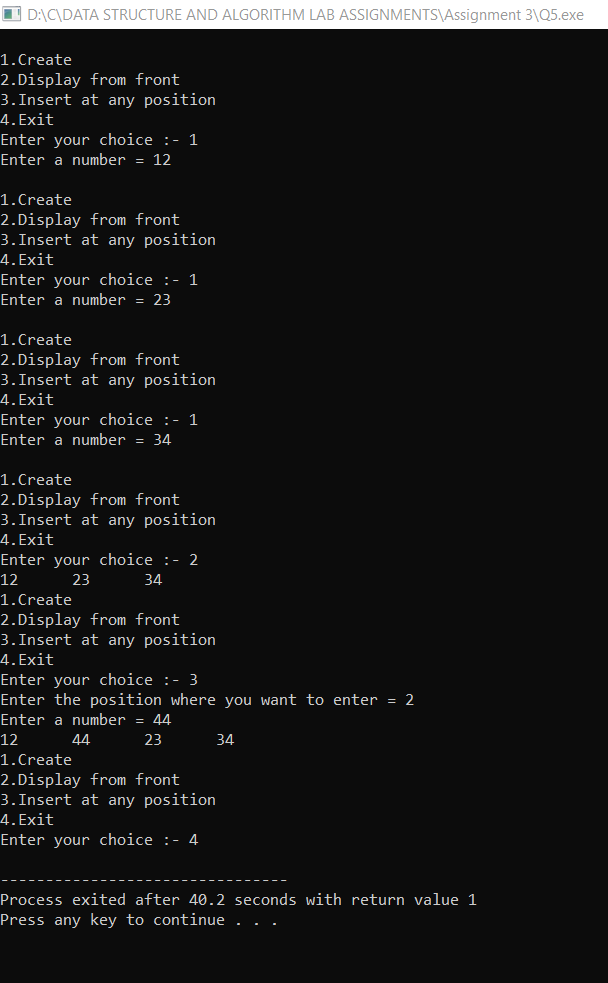
ptr->prev=nw;

}

display\_from\_front();

}

**Output:-**

****

**Program 6:-**

**Output:-**

**Program 7:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void split();

void display\_first();

void display\_second();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last, \*last1, \*head1;

int c = 0;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Split the list into 2 equal halves\n4.Display 1st half\n5.Display 2nd half\n6.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

c++;

break;

case 2:

display\_from\_front();

break;

case 3:

split();

break;

case 4:

display\_first();

break;

case 5:

display\_second();

break;

case 6:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void split()

{

int i = 0;

head1=NULL;

last1=NULL;

node \*ptr,\*ptr1;

while(i!=(c/2))

{

ptr1=ptr;

ptr=ptr->next;

}

ptr1->next=NULL;

last1=ptr1;

ptr->prev=head;

head1=ptr;

}

void display\_first()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!");

else

{

ptr = head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void display\_second()

{

node \*ptr;

if(head1==NULL)

printf("Linked list does not exists!!");

else

{

ptr = head1;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

**Output:-**

**Program 8:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void insert\_at\_any();

typedef struct Node{

char s[20];

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

fflush(stdin);

nw = (node \*)malloc(sizeof(node));

printf("Enter a string = ");

gets(nw->s);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%s\n",ptr->s);

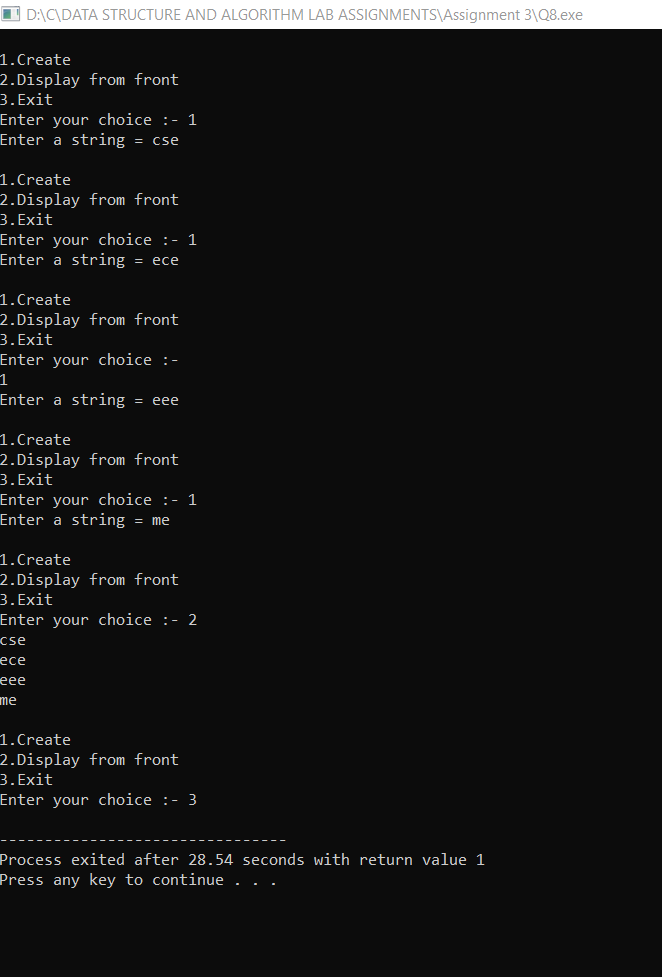
ptr = ptr->next;

}

}

}

**Output:-**

****

**Program 9:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void insert\_at\_suit();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

printf("\nEnter the elements in the nodes in ascending order");

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Insert at suitable position\n4.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

insert\_at\_suit();

break;

case 4:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void insert\_at\_suit()

{

node \*nw,\*ptr,\*ptr1;

nw=(node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

if(head==NULL)

printf("Linked List does not exists!!");

else

{

ptr = head;

ptr1= head;

while(nw->data>ptr->data&&ptr!=NULL)

{

ptr1 = ptr;

ptr = ptr->next;

}

ptr1->next=nw;

nw->next=ptr;

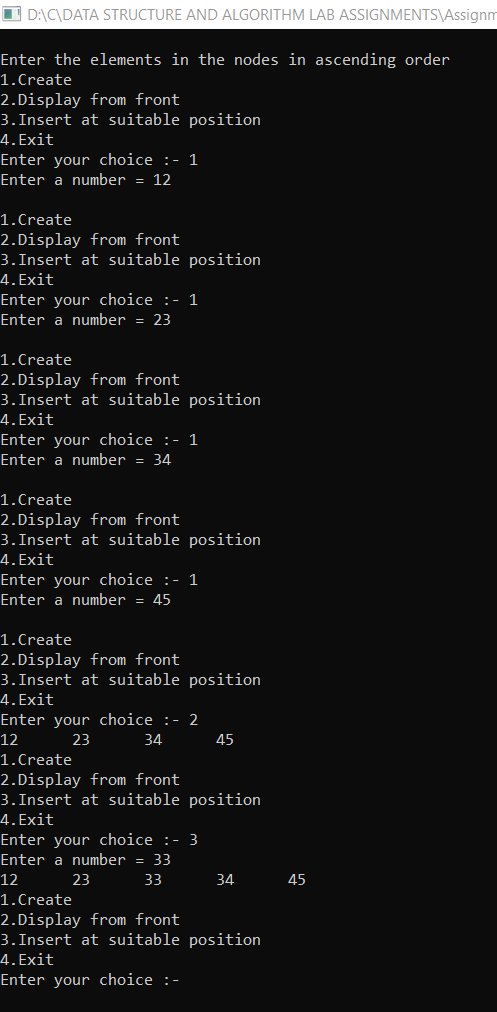
ptr->prev=nw;

nw->prev=ptr1;

}

display\_from\_front();}

**Output:-**

****

**Program 10:-**

#include<stdio.h>

#include<stdlib.h>

void create();

void display\_from\_front();

void exchange();

typedef struct Node{

int data;

struct Node \*next;

struct Node \*prev;

}node;

node \*head, \*last;

int main()

{

int ch;

head = NULL;

last = NULL;

while(1)

{

printf("\n1.Create\n2.Display from front\n3.Exchange first and last node\n4.Exit\nEnter your choice :- ");

scanf("%d",&ch);

switch(ch)

{

case 1:

create();

break;

case 2:

display\_from\_front();

break;

case 3:

exchange();

break;

case 4:

exit(1);

break;

default:

printf("\nINVALID CHOICE!!!!!!");

}

}

}

void create()

{

node \*nw, \*ptr;

nw = (node \*)malloc(sizeof(node));

printf("Enter a number = ");

scanf("%d",&nw->data);

nw->next = NULL;

nw->prev = NULL;

if(head==NULL&&last==NULL)

{

head=nw;

last = head;

}

else

{

ptr = last;

ptr->next=nw;

nw->prev=ptr;

last = nw;

}

}

void display\_from\_front()

{

node \*ptr;

if(head==NULL)

printf("Linked list does not exists!!!1");

else

{

ptr=head;

while(ptr!=NULL)

{

printf("%d\t",ptr->data);

ptr = ptr->next;

}

}

}

void exchange()

{

int temp;

temp = head->data;

head->data=last->data;

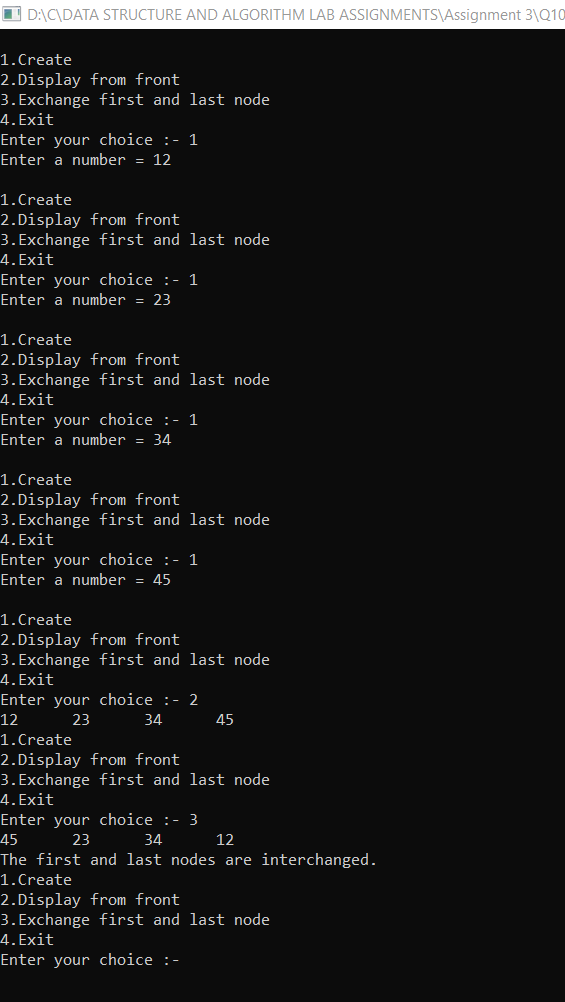
last->data=temp;

display\_from\_front();

printf("\nThe first and last nodes are interchanged.");

}

**Output:-**

****